

HS  
Docket No. 1293.1127 (MDS)



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of: )  
)  
Jae-seong SHIM )  
) Group Art Unit: Unassigned  
Serial No.: To be assigned )  
) Examiner: Unassigned  
Filed: July 20, 2000 )

For: SCRAMBLER AND SCRAMBLING METHOD

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN**  
**APPLICATION IN ACCORDANCE**  
**WITH THE REQUIREMENTS OF 37 C.F.R. §1.55**

*Honorable Commissioner of  
Patents and Trademarks  
Washington, D.C. 20231*

*Sir:*

In accordance with the provisions of 37 C.F.R. §1.55, the applicant submits herewith a certified copy of the following foreign application:

Korean Patent Application No. 99-29280  
Filed: 20 July 1999

It is respectfully requested that the applicants be given the benefit of the foreign filing date as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. §119.

Respectfully submitted,

STAAS & HALSEY LLP

By: 

Michael D. Stein

Registration No. 37,240

July 20, 2000  
700 11th Street, N.W., Ste. 500  
Washington, D.C. 20001  
(202) 434-1500

jc784 U.S. PTO  
09/620462  
07/20/00



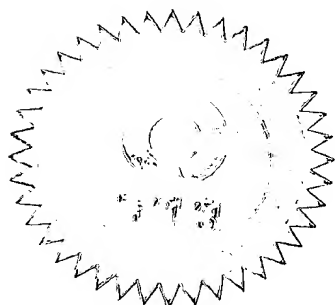
별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Industrial  
Property Office.

출원 번호 : 특허출원 1999년 제 29280 호  
Application Number,

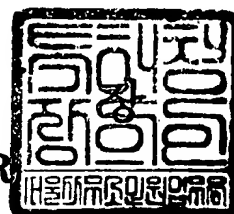
출원 년 월 일 : 1999년 07월 20일  
Date of Application

출원인 : 삼성전자 주식회사  
Applicant(s)



2000 년 04 월 21 일

특 허 청  
COMMISSIONER



**【서류명】** 특허출원서  
**【권리구분】** 특허  
**【수신처】** 특허청장  
**【참조번호】** 0009  
**【제출일자】** 1999.07.20  
**【국제특허분류】** G11B  
**【발명의 명칭】** 스크램블러와 스크램블 방법  
**【발명의 영문명칭】** Scrambler and scramble method  
**【출원인】**

**【명칭】** 삼성전자 주식회사  
**【출원인코드】** 1-1998-104271-3

**【대리인】**

**【성명】** 이영필  
**【대리인코드】** 9-1998-000334-6  
**【포괄위임등록번호】** 1999-009556-9

**【대리인】**

**【성명】** 권석흠  
**【대리인코드】** 9-1998-000117-4  
**【포괄위임등록번호】** 1999-009576-5

**【대리인】**

**【성명】** 이상용  
**【대리인코드】** 9-1998-000451-0  
**【포괄위임등록번호】** 1999-009577-2

**【발명자】**

**【성명의 국문표기】** 심재성  
**【성명의 영문표기】** SHIM, Jae Seong  
**【주민등록번호】** 641223-1058515  
**【우편번호】** 143-191  
**【주소】** 서울특별시 광진구 자양1동 229-24  
**【국적】** KR

**【취지】**

특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대  
 리인 이영  
 필 (인) 대리인  
 권석흠 (인) 대리인  
 이상용 (인)

**【수수료】**

【기본출원료】 20 면 29,000 원

【가산출원료】 15 면 15,000 원

【우선권주장료】 0 건 0 원

【심사청구료】 0 항 0 원

【합계】 44,000 원

【첨부서류】 1. 요약서·명세서(도면)\_1통

**【요약서】****【요약】**

본 발명에는 스크램블러와 스크램블 방법이 개시되어 있다. 본 발명의 스크램블러는 랜덤 데이터 발생 주기를 최소한 제1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 계산한 결과를 승산하여 얻어지는 결과를 토대로 하여 랜덤 데이터를 발생하는 랜덤 데이터 발생기를 포함하여, 고밀도 디스크 시스템에서 안정된 서보 신호 생성 및 변조시 DC 억압에 유리한 스크램블러를 제공한다.

**【대표도】**

도 4

**【명세서】****【발명의 명칭】**

스크램블러와 스크램블 방법{Scrambler and scramble method}

**【도면의 간단한 설명】**

도 1은 일반적인 디스크의 내주와 외주의 반지름을 보인 도면이다.

도 2는 일반적인 DVD 시스템의 스크램블러의 회로도이다.

도 3은 도 2에 도시된 레지스터에서 사용되는 초기값을 보인 테이블이다,

도 4는 본 발명에 의한 고밀도 디스크 시스템을 위한 스크램블러의 일 실시예에 따른 회로도이다.

도 5는 도 4에 도시된 8 비트 시프트-스크램블러에서 사용되는 레지스터의 초기값을 보인 테이블이다.

도 6은 본 발명의 이해를 돕기 위한 고밀도 디스크 시스템을 위한 스크램블러를 일 반화한 회로도이다.

도 7은 도 6에 도시된 랜덤 데이터 발생기의 랜덤 데이터의 주기가 64K이고, 유효 브랜치의 수가 4일 때 브랜치값을 정리한 테이블이다.

도 8은 본 발명에 의한 고밀도 디스크 시스템을 위한 1비트 시프트-스크램블러에서 사용되는 레지스터의 초기값을 보인 테이블이다.

도 9는 본 발명에 의한 고밀도 디스크 시스템을 위한 스크램블러의 다른 실시예에 따른 회로도이다.

도 10은 도 9에 도시된 스크램블러의 구조를 4K 단위로 변경하기 위한 제어값을 정리한 테이블이다.

#### 【발명의 상세한 설명】

#### 【발명의 목적】

#### 【발명이 속하는 기술분야 및 그 분야의 종래기술】

- <11> 본 발명은 데이터 스크램블 분야에 관한 것으로, 특히 고밀도 디스크 시스템에 적합한 스크램블러와 스크램블 방법에 관한 것이다.
- <12> 일반적으로 데이터 스크램블의 목적은 키(key)를 갖고 있지 않은 사용자로부터 데이터를 보호하기 위한 수단으로 많이 사용되고 있으며, 통신용으로는 보호 통신(security communication) 목적으로 널리 사용되고 있는 랜덤화의 한 방법이다.
- <13> 광 디스크 예를 들어, CD-ROM(Compact Disc Read Only Memory) 또는 DVD(Digital Versatile Disc) 등을 사용하는 광 디스크 시스템의 스크램블러(scrambler)내에 입력되는 특정값의 데이터를 난수로 만들어 주는 랜덤 데이터 발생기를 채용하고 있다.
- <14> 광 디스크 시스템에서 입력 데이터를 스크램블하는 주요한 첫 번째 이유로는 차동 위상 검출(Differential Phase Detection: DPD)을 이용한 트래킹 제어를 원활히 수행하기 위함이다. 만일, 동일한 데이터가 입력되어 동일한 변조 코드들이 디스크 상의 인접 트랙에 기록되면 재생시 DPD 신호가 검출되지 않아 서보부(servo unit)에서 트래킹 제어가 어려워진다. 예를 들어, 스크램블하지 않는 CD-Audio의 경우 곡과 곡사이 구간(데이터 모두 '00h'구간)에서 DPD 제어가 어렵다.
- <15> 두 번째 이유는 변조부에서 DC 억압 제어에 부담을 줄여주기 위해서이다. 동

일한 데이터가 연속 입력되는 경우 특정한 값에 대해서는 DSV(Digital Sum Value) 제어 자체가 불가능할 수도 있다. 이러한 최악의 경우를 막기 위해서 데이터의 랜덤화가 필요하다. 여기서, DSV는 코드워드의 DC 방향을 예측하는 파라미터로서, 변조 코드워드가 DC에 수렴하는 특성을 갖는 것이 바람직하다.

<16> 세 번째 이유는 특정 데이터를 보호하기 위함이다. CD-ROM(Read Only Memory)의 경우는 데이터내의 싱크 패턴(00h, FFh, FFh, ..., FFh, 00h)을 보호하기 위하여 싱크를 제외한 나머지 데이터만 스캔램블을 수행한다.

<17> 일반적인 DVD 시스템을 위한 스캔램블러의 주기성을 도 1을 결부시켜 설명한다. 채널 비트의 길이는  $0.133\mu\text{m}$ 이고, 1 섹터의 물리적 길이는  $5.146\text{mm}(= 0.133\mu\text{m} \times 1488 \times 26)$ 이고, 최내주 반지름은 도 1에 도시된 바와 같이  $24\text{mm}$ 이므로 최내주 트랙길이는  $150.8\text{mm}(= 2\pi r)$ 이고, 최내주 트랙용량은 29.3섹터( $= 150.8\text{mm}/5.146\text{mm}$ )이다. 또한, 최외주 반지름은 도 1에 도시된 바와 같이  $58\text{mm}$ 이므로 최외주 트랙길이는  $364.42\text{mm}(= 2\pi r)$ 이고, 최외주 트랙 용량은 70.82섹터( $= 364.42\text{mm}/5.146\text{mm}$ )이다.

<18> DPD 제어를 위해서 스캔램블러의 랜덤 데이터 발생 주기가 최외주에서는 최소 141.64 섹터( $= 70.82 \text{ 섹터} \times 2$ ) 이상이 되어야 하고, 최내주에서는 29.3 섹터 이내에서 동일한 데이터 반복되더라도 DPD 제어에는 문제가 발생하지 않는다.

<19> 도 2는 일반적인 DVD 시스템을 위한 스캔램블러의 회로도로서, 배타적 논리합 게이트(10)와 랜덤 데이터를 제공하는 레지스터( $r_0 \sim r_{14}$ )를 랜덤 데이터 발생기라고 하고, 이 랜덤 데이터 발생기와 논리합 게이트들(11~18)을 포함해서 스캔램블러라고 한다.

<20> 도 2에 있어서, 15 비트의 레지스터( $r_0 \sim r_{14}$ )는 도면에는 도시되지 않았지만 스캔



램블을 위한 클럭 입력에 동기해서 시프트 레프트(Shift Left)하며, 최하위 레지스터( $r_0$ )의 입력은 배타적 논리합(Exclusive OR:XOR) 게이트(10)로부터 제공되는 최상위 레지스터( $r_{14}$ )의 출력과 11번째 하위 레지스터( $r_{10}$ )의 출력을 배타적 논리합한 값이 된다.

<21> 도 2에 도시된 랜덤 데이터 발생기의 랜덤 데이터 발생 주기는 32K(kilo)이며, 이 랜덤 데이터 발생 주기는 DVD의 1 ECC(Error Correction Code) 블록의 크기인 32K와 일치한다. 즉, 1 ECC 블록내에서는 주기성이 없는 랜덤 데이터를 발생시키며, 레지스터( $r_0 \sim r_{14}$ )를 8번 시프트 레프트(shift left)한 후 XOR 게이트들(11~18)에서 8개의 하위 레지스터( $r_0 \sim r_7$ )의 출력과 1 바이트의 입력 데이터( $D_0 \sim D_7$ )를 배타적 논리합함으로써 스크램블된 결과가 얻어진다. 여기서, 도면에는 도시되지 않았지만 XOR 게이트(11~18)를 위한 데이터 클럭 속도는 레지스터( $r_0 \sim r_{14}$ )를 위한 스크램블 클럭 속도보다 1/8의 속도를 갖는다.

<22> 한편, 레지스터( $r_0 \sim r_{14}$ )를 8번 시프트 레프트한 후 스크램블을 하기 때문에 매 섹터마다 할당된 4 바이트의 식별코드(ID) 중 마지막 1 바이트의 상위 4 비트(ID(7:4))를 참조하여 초기값(preset value)으로 레지스터( $r_0 \sim r_{14}$ )를 초기화시키는 데 이때 초기값을 선정하는 데 주의를 해야 한다. 즉, 동일한 데이터가 입력되더라도 1 섹터 내에서는 초기값으로부터 랜덤 데이터가 발생하며 이 1 섹터내의 랜덤 데이터들은 1 ECC 블록(16섹터)동안 동일한 초기값에 의해 반복된다.

<23> 도 3에 도시된 바와 같이, 최초의 레지스터( $r_0 \sim r_{14}$ )의 초기값 '0001h'와 '0001h'를 7번까지 시프트 레프트한 값들(0002h, 0004h, 0008h, 0010h, 5000h, 2001h, 4002h)과 이들 값이 리턴하는 데 필요한 용량 16K(=2K

$\times 8$ )이후의 레지스터( $r_0 \sim r_{14}$ )의 값인 '5500h'와 '5500h'를 7번까지 시프트 레프트한 값들(2A00h, 5400h, 2800h, 5000h, 2001h, 4002h, 0005h)을 레지스터( $r_0 \sim r_{14}$ )의 초기값으로 사용하고 있다.

<24> 도 2에 도시된 스크램블러는 랜덤 데이터 발생기에서 발생된 32K의 랜덤 데이터를 모두 사용하고, 1 ECC 블록내의 섹터 데이터들은 반복하지만 도 1에서 언급한 바와 같이 DPD 제어에는 문제가 없고, 또한 변조시 1 섹터 동안은 랜덤 데이터를 발생하므로 DSV 제어에도 문제가 되지 않는다. 도 3에 도시된 바와 같은 레지스터의 초기값 방법을 이용하여 256 섹터(=1 ECC 블록(16섹터)  $\times$  16회의 초기화) 동안은 ECC 블록간에 동일 데이터가 연속적으로 발생하지 않는다. 따라서, 디스크의 최외주에서도 연속한 트랙에서 동일한 코드스트림이 발생하지 않으므로 DPD 제어에 문제가 없다.

<25> 그러나, 종래의 랜덤 데이터 발생기와 이를 사용하는 스크램블러는 32K보다 큰 주기의 랜덤 데이터 발생과 스크램블을 필요로 할 때 대응이 안되는 문제점이 있었다.

【발명이 이루고자 하는 기술적 과제】

<26> 따라서, 본 발명의 목적은 랜덤 데이터의 발생 주기를 제어하여 고밀도 광디스크 시스템에 적합한 스크램블러와 스크램블 방법을 제공하는 데 있다.

<27> 본 발명의 다른 목적은 안정된 서보 신호 생성 및 변조시 DC 억압에 유리한 스크램블러와 스크램블 방법을 제공하는 데 있다.

<28> 상기한 목적들을 달성하기 위하여, 본 발명에 의한 스크램블러는 데이터 스크램블러에 있어서: 랜덤 데이터 발생 주기를 최소한 제1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 제산한 결과를 승산하여 얻어진

결과를 토대로하여 랜덤 데이터를 발생하는 랜덤 데이터 발생기를 가지는 것을 특징으로 하고 있다.

<29> 또한, 본 발명에 의한 스크램블 방법은 랜덤 데이터 발생기를 이용하여 데이터를 스크램블하는 방법에 있어서: 랜덤 데이터 발생기의 랜덤 데이터 발생 주기를 최소한 제 1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 계산한 결과를 승산하여 얻어진 결과를 토대로하여 랜덤 데이터를 발생하는 단계를 포함함을 특징으로 하고 있다.

#### 【발명의 구성 및 작용】

<30> 이하, 첨부된 도면을 참조하여 본 발명에 의한 스크램블러와 스크램블 방법의 바람직한 실시예를 설명하기로 한다.

<31> 먼저, 고밀도(High Density:HD) DVD 시스템의 스크램블러의 주기성을 살펴보면 다음과 같다.

<32> HD-DVD는 일반적인 DVD와 비교하여 최내주와 최외주는 동일하고 선밀도가 2배로 높다고 가정하면, 최내주 트랙 길이는  $150.8\text{mm}(= 2\pi 24\text{mm})$ 이고, 최내주 트랙 용량은 약  $120\text{KB}(= 60 \times 2\text{KB})$ 이고, 최외주 트랙 길이는  $364.42\text{mm}(= 2\pi 58\text{mm})$ 이고, 최외주 트랙 용량은 약  $284\text{KB}(= 142 \times 2\text{KB})$ 이다.

<33> HD-DVD의 선밀도를 DVD 대비 2배로 높다고 가정하므로 DPD 제어를 위해서 최외주에서는 스크램블러의 랜덤 데이터 발생 주기가 최소  $564\text{K}(= 284\text{K} \times 2)$  이상이 되어야 하고, 최내주에서는  $120\text{K}$ 이내에서 동일한 데이터가 반복되더라도 DPD 제어에는 문제가 발생하지 않음을 알 수 있다.

- <34> 그러나, HD-DVD에서는 1 섹터의 크기를 2KB로 하느냐 또는 4KB로 하느냐와 1 ECC 블록의 크기를 32 섹터로 할 것인지 또는 16 섹터로 할 것인지에 따라 스크램블러의 구조가 바뀌게 되며, 이에 대해서 보다 상세히 설명하기로 한다.
- <35> 먼저, 1 섹터가 2KB이고, 1 ECC 블록이 32 섹터일 경우의 HD-DVD 시스템을 위한 스크램블러의 구조는 도 2에 도시된 일반적인 DVD 시스템을 위한 스크램블러를 사용할 수 있다.
- <36> 즉, 최내주의 트랙 용량인 120K는 1 ECC 블록(64KB)보다는 크고, 2 ECC 블록(128KB)보다는 작으므로 DVD와 동일하게 1 ECC 블록동안 1 섹터마다 동일한 값의 초기값 설정이 가능하다. 최외주의 두 개의 트랙동안 동일한 데이터가 반복되지 않게 하기 위해서는 564K 이상의 주기성 확보를 해야하는 데 16개의 초기값만 있으면 랜덤 데이터 발생 주기가  $1024KB(=1 \text{ ECC 블록}(64KB) \times 16)$ 가 되어 최외주에서의 최소 랜덤 데이터 발생 주기 564KB를 초과하므로 문제가 없다. 따라서, 도 2에 도시된 바와 같은 일반적인 DVD 시스템을 위한 스크램블러와 동일한 구조로도 HD-DVD 시스템에서 사용이 가능하다.
- <37> 다음, 1 섹터는 4KB이고, 1 ECC 블록은 16 섹터일 경우의 스크램블러의 구조는 일반적인 DVD 시스템을 위한 스크램블러를 사용하지 못하고, 스크램블러의 구조가 변경되어야 한다.
- <38> 최내주에서는 1 ECC블록(64KB)보다는 크고, 2 ECC 블록(128KB)보다는 작으므로 1 ECC 블록동안 1 섹터마다 동일한 값의 초기값 설정이 가능하다.
- <39> 최외주에서도 564K 이상의 주기성을 확보하기 위해 16개의 초기값만 있으면 랜덤 데이터 발생 주기가  $1024KB(=1 \text{ ECC}(64KB) \times 16)$ 가 되어 최소 랜덤 데이터 발생 주기

564KB를 초과하므로 문제가 없다. 그러나, 스크램블러의 랜덤 데이터 발생기의 랜덤 데이터 발생 주기는  $64KB (= 1 \text{ 섹터}(4KB) \times 16)$  이상이어야 하므로 도 2에 도시된 바와 같은 일반적인 DVD의 스크램블러의 구조로는 사용이 불가능하다.

<40> 따라서, 본 발명에서는 랜덤 데이터 발생 주기가 64K 이상이 되는 3가지 형태의 스크램블러 구조를 제안한다.

<41> 본 발명에서 제안하는 스크램블러의 첫 번째 구조는 도 4에 도시되어 있으며, HD-DVD 시스템을 위한 스크램블러의 구조는 도 2에 도시된 바와 같은 일반적인 DVD 시스템을 위한 스크램블러 구조와 유사한 구조이다.

<42> 즉, 16개의 레지스터( $r_0 \sim r_{15}$ )로부터 16비트의 랜덤 데이터가 발생되고, XOR 게이트들(111~118)로부터 1 바이트의 입력 데이터( $D_0 \sim D_7$ )와 8개의 하위 레지스터( $r_0 \sim r_7$ )의 각 출력을 배타적 논리합한 스크램블 결과가 제공된다. XOR 게이트(103)는 최상위 레지스터( $r_{15}$ )의 출력과 그 다음 상위 레지스터( $r_{14}$ )의 출력을 배타적 논리합하고, XOR 게이트(102)는 XOR 게이트(103)의 출력과 상위 레지스터( $r_{13}$ )의 출력을 배타적 논리합하고, XOR 게이트(101)는 XOR 게이트(102)의 출력과 레지스터( $r_{10}$ )의 출력을 배타적 논리합해서 최하위 레지스터( $r_0$ )에 피드백 입력하고 있다.

<43> 도 4에 도시된 HD-DVD 시스템을 위한 8 비트 시프트-스크램블러에서 사용되는 레지스터의 초기값을 보인 테이블은 도 5에 도시된 바와 같다. 스크램블은 각 레지스터( $r_0 \sim r_{15}$ )의 출력을 8 비트 시프트 레프트한 후 레지스터( $r_0 \sim r_7$ )의 하위 8 비트와 1 바이트의 입력 데이터( $D_0 \sim D_7$ ) 단위로 수행되므로 레지스터( $r_0 \sim r_{15}$ )의 초기값은 최초 0001h와 이를 7회까지 시프트 레프트한 값들(0002h, 0004h, 0008h,

0010h, 0020h, 0040h, 0080h)과 이들 값이 리턴하는 데 필요한 용량  $32K(4K \times 8\text{회})$  이후의 레지스터( $r_0 \sim r_{15}$ )의 결과인 7E80h와 이를 7회까지 시프트 레프트한 값들(FF01h, FE02h, FC04h, F808h, F011h, E023h, C046h)을 레지스터( $r_0 \sim r_{15}$ )를 위한 섹터 단위의 초기값으로 사용한다.

<44> 여기서, 도 4에 도시된 바와 같은 64K의 랜덤 데이터 발생 주기를 가지는 랜덤 데이터 발생기를 채용한 스크램블러의 다양한 예들은 동출원인에 의해 '랜덤 데이터 발생기와 이를 사용하는 스크램블러'라는 제목으로 199년 7월 10일에 출원한 대한민국 출원번호 제 99-27886호에 개시되어 있다.

<45> 본 발명의 이해를 돕기 위해 상기 출원서에 언급되어 있는 도 4에 도시된 스크램블러의 일반화된 구조를 도 6 및 도 7을 결부시켜 설명하기로 한다.

<46> 도 6에 도시된 스크램블러의 브랜치 테이블(200)은 일 예로서 도 7에 도시된 바와 같이 랜덤 데이터 발생기의 XOR 게이트( $G_0 \sim G_{15}$ )의 유효 브랜치의 수가 4인 경우 모든 가능한 경우의 브랜치값( $D_{00} \sim D_{015}$ )이 저장되어 있다. 도 4에 도시된 스크램블러는 브랜치의 값이 'B400h'인 경우에 대해 간단한 구조로 구현한 것이다.

<47> 도 6에 있어서, 각 멀티플렉서( $m_0 \sim m_{15}$ )는 브랜치 테이블(200)에 저장된 브랜치값 중 어느 하나가 선택되면 선택된 브랜치값에 따른 16비트의 출력( $D_{00} \sim D_{015}$ )을 선택 신호로 입력해서, 브랜치 테이블(200)의 출력( $D_{00} \sim D_{015}$ )의 값이 '1'일 때는 제1 입력단(A)으로 입력되는 '0'를 출력 신호( $M_{00}$

$\sim Mo_{15}$ )로서 각 XOR 게이트( $G_0 \sim G_{15}$ )의 각 일단에 제공한다. 이때, 각 XOR 게이트( $G_0 \sim G_{15}$ )는 각 타단으로 입력되는 각 레지스터( $r_0 \sim r_{15}$ )의 출력( $S_0 \sim S_{15}$ )을 그대로 출력하게 되고, 결국 누적된 XOR 게이트( $G_0$ )의 출력값이 최하위 레지스터( $r_0$ )에 피드백 입력된다.

<48> 또한, 각 멀티플렉서( $m_0 \sim m_{15}$ )는 브랜치 테이블(200)의 출력( $Do_0 \sim Do_{15}$ )의 값이 '0' 일 때는 각 제2 입력단(B)으로 입력되는 레지스터( $r_0 \sim r_{15}$ )의 출력( $S_0 \sim S_{15}$ )을 자신의 출력( $Mo_0 \sim Mo_{15}$ )으로 제공하면, 각 XOR 게이트( $G_0 \sim G_{15}$ )는 각 멀티플렉서( $m_0 \sim m_{15}$ )의 출력인  $S_0 \sim S_{15}$ 과 각 레지스터( $r_0 \sim r_{15}$ )의 출력인  $S_0 \sim S_{15}$  자신과 배타적 논리합한다. 이때, 각 XOR 게이트( $G_0 \sim G_{15}$ )의 출력이 결국 '0'이 되어 최하위 레지스터( $r_0$ )에 피드백 입력되는 값은 없다.

<49> 이렇게 각 레지스터( $r_0 \sim r_{15}$ )로부터 1 ECC 블록동안 64K의 랜덤 데이터가 발생되고, XOR 게이트들(201~208)로부터 1 바이트의 입력 데이터( $D_0 \sim D_7$ )와 하위 8개의 레지스터( $r_0 \sim r_7$ )의 각 출력을 배타적 논리합한 스크램블 결과가 제공된다.

<50> 여기서, 레지스터의 수를 16으로 하면 랜덤 데이터의 주기는  $2^{16}(=64K)$ 가 되고, 레지스터의 수를  $n$ 으로 하면 랜덤 데이터의 주기는  $2^n$ 으로 확장할 수 있다.

<51> 본 발명에서 제안하는 스크램블러의 두 번째 구조는 도 4 또는 도 6에 도시된 스크램블러와 동일한 구조를 갖지만 다만 스크램블이 1비트마다 시프트 레프트한 후 레지스터( $r_0$

$\sim r_7$ )의 하위 8 비트와 1 바이트의 입력 데이터 단위로 이루어지기 때문에 매 섹터마다 할당된 4 바이트의 식별코드(ID) 중 마지막 1 바이트의 상위 4 비트(ID(7:4))의 값에 따른 레지스터( $r_0 \sim r_{15}$ )의 초기값은 도 8에 도시된 바와 같이 도 5에 도시된 초기값과는 다른 값을 갖는다. 도 8에 도시된 초기값은 도 4에 도시된 스크램블에서 사용되는 레지스터의 초기값이다.

<52> 즉, 최초 0001h와 이로부터 매 4096회 시프트 레프트한 후의 얻어지는 15개의 레지스터( $r_0 \sim r_{15}$ )의 값들(3DADh, D4E7h, FDCAh, EBCCh, 292Eh, 50F0h, BFCAh, 7F80h, D36Eh, BB39h, 5DFFh, A809h, 6647h, 8044h, 0304h)을 초기값으로 사용하다는 점이 다르다.

<53> 본 발명에서 제안하는 스크램블러의 두 번째 구조는 높은 속도의 요구에 따른 대책으로서 패러렐 구조로 변경해야 하는 번거로움이 없다는 장점이 있다.

<54> 본 발명에서 제안하는 스크램블러의 세 번째 구조는 도 9에 도시되어 있으며, 도 4 또는 도 6에 도시된 스크램블러의 구조는 64K 주기의 랜덤 데이터를 생성하는 레지스터( $r_0 \sim r_{15}$ )를 구비하고 매 섹터마다 초기값을 바꾸면서 스크램블을 수행하는 반면, 도 9에 도시된 스크램블러는 4K 주기의 랜덤 데이터를 생성하는 레지스터를 16 종류를 두어 매 섹터마다 할당된 4 바이트의 식별코드(ID) 중 마지막 1 바이트의 상위 4 비트(ID(7:4))의 값에 따라 랜덤 데이터 발생기의 구조 자체를 바꾸는 형태이다.

<55> 도 9에 도시된 스크램블러는  $4 \times 16$  디코더(300), 12개의 멀티플렉서( $m_0 \sim m_{11}$ ), 12개의 XOR 게이트( $G_0 \sim G_{11}$ ), 12개의 레지스터( $r_0 \sim r_{11}$ )를 포함하는 랜덤 데이터 발생기와 1 바이트의 입력 데이터( $D_0 \sim D_7$ )와 8개의 하위 레지스터( $r_0 \sim r_7$ )의 각 출력을 배타적 논리합하여 스크램블 결과를 출력하는 XOR 게이트들(301



~308)로 구성되어 있다. 여기서, 12개의 멀티플렉서를  $4 \times 16$  디코더(300)의 12비트의 출력에 따라 '0' 또는 각 레지스터의 출력을 선택출력하는 선택출력회로로, 12개의 XOR 게이트를 12비트의 배타적 논리합 결과를 제공하는 논리 회로로 지칭될 수 있다.

<56> 도 9에 있어서,  $4 \times 16$  디코더(300)는 도 10에 도시된 바와 같이 매 섹터마다 할당된 4 바이트의 식별 코드 중 마지막 1 바이트의 상위 4 비트(ID(7:4))의 값에 따른 16개의 스크램블러 구조의 변경을 제어하는 제어값(829h, 834h, 84Ch, 868h, 883h, 891h, 8B0h, 8C2h, 906h, 960h, 990h, A03h, A18h, B04h, C48h, CA0h)중 하나를 1 ECC 블록마다 입력하면 이 제어값에 대응하는 12비트의 출력을 제공하며, 12 비트의 출력(Do<sub>0</sub>~Do<sub>11</sub>) 중 '1'에 해당하는 값을 가지는 브랜치만 유효하다. 여기서, 레지스터(r<sub>0</sub>~r<sub>11</sub>)의 초기값은 하나의 ECC 블록 단위로 설정되며, 일 예로 '001h'로 설정한다.

<57> 각 멀티플렉서(m<sub>0</sub>~m<sub>11</sub>)는  $4 \times 16$  디코더(300)의 12비트의 출력(Do<sub>0</sub>~Do<sub>11</sub>)을 선택 신호로 입력해서,  $4 \times 16$  디코더(300)의 12비트의 출력(Do<sub>0</sub>~Do<sub>11</sub>)의 값이 '1'일 때는 제1 입력단(A)으로 입력되는 '0'를 출력 신호(Mo<sub>0</sub>~Mo<sub>11</sub>)로서 각 XOR 게이트(G<sub>0</sub>~G<sub>11</sub>)의 각 일단에 제공한다. 이때, 각 XOR 게이트(G<sub>0</sub>~G<sub>11</sub>)는 각 타단으로 입력되는 각 레지스터(r<sub>0</sub>~r<sub>11</sub>)의 출력(S<sub>0</sub>~S<sub>11</sub>)을 그대로 출력하게 되고, 결국 누적된 XOR 게이트(G<sub>0</sub>~G<sub>11</sub>)의 출력값이 최하위 레지스터(r<sub>0</sub>)에 피드백 입력된다.

<58> 또한, 각 멀티플렉서(m<sub>0</sub>

$\sim m_{11}$ )는  $4 \times 16$  디코더(300)의 12비트의 출력( $Do_0 \sim Do_{11}$ )의 값이 '0'일 때는 각 제2 입력 단(B)으로 입력되는 레지스터( $r_0 \sim r_{11}$ )의 출력( $S_0 \sim S_{11}$ )을 자신의 출력( $Mo_0 \sim Mo_{11}$ )으로 제공한다, 각 XOR 게이트( $G_0 \sim G_{11}$ )는 각 멀티플렉서( $m_0 \sim m_{11}$ )의 출력인  $S_0 \sim S_{11}$ 과 각 레지스터( $r_0 \sim r_{11}$ )의 출력인  $S_0 \sim S_{11}$  자신과 배타적 논리합한다. 이때, 각 XOR 게이트( $G_0 \sim G_{11}$ )의 출력이 결국 '0'이 되어 최하위 레지스터( $r_0$ )에 피드백 입력되는 값은 없다.

<59> 이렇게 각 레지스터( $r_0 \sim r_{11}$ )로부터 매 섹터마다 4K의 랜덤 데이터가 발생되고, XOR 게이트들(301~308)로부터 1 바이트의 입력 데이터( $D_0 \sim D_7$ )와 8개의 하위 레지스터( $r_0 \sim r_7$ )의 각 출력을 배타적 논리합한 스크램블 결과가 제공된다.

<60> 결론적으로, 스크램블러의 구조는 최내주 트랙과 최외주 트랙의 용량, 섹터 크기 및 ECC 블록의 크기에 따라 변경 가능하다. 즉, 섹터 크기가 2KB이고, ECC 블록의 크기가 32 섹터인 HD-DVD를 사용하는 시스템에서는 현재 DVD 시스템에서 사용중인 스크램블러를 그대로 사용 가능하다. 그러나, 섹터의 크기를 4KB로 하고 ECC 블록의 크기를 16 섹터로 하는 HD-DVD를 사용하는 시스템에서는 아래의 3가지 형태 i), ii), iii) 중 하나를 사용가능하다.

<61> i) 초기값을 갖는 랜덤 데이터 발생기 + 8 비트 시프트-스크램블러: 도 4(도 6) + 도 5

<62> ii) 초기값을 갖는 랜덤 데이터 발생기 + 1비트 시프트-스크램블러: 도 4(도 6) + 도 8

<63> iii) 구조 변경이 가능한 랜덤 데이터 발생기 + 스크램블러: 도 9 + 도 10

<64> 한편, 광 디스크 시스템을 위한 스크램블러내에 채용된 랜덤 데이터 발생기의 랜덤 데이터 발생 주기 조건을 알아보면 다음과 같다.

<65> 제1 데이터 프레임(Data Frame 1)은 섹터이고, 제2 데이터 프레임(Data Frame 2)은 ECC 블록이고, 제1 데이터 프레임내의 데이터 양을  $b$ 라고 하고, 제2 데이터 프레임내의 데이터 양을  $B$ 라고 하고, 최내주 트랙내의 데이터 양은  $A$ 라고 하고, 최외주 두 개의 트랙내의 데이터 양을  $C$ 라고 하면 아래의 조건 1, 조건 2, 조건 3을 만족하며, 광 디스크 시스템의 스크램블러내의 랜덤 데이터 발생기의 랜덤 데이터 발생 주기는 최소  $b \times C/B$  이상이어야 한다.

<66> 조건 1)  $\text{Data Frame 2} = n \times \text{Data Frame 1}$ ,  $n$ =정수

<67> 조건 2)  $\|A/B\| = \alpha$ ,  $\|A/B\|$  는  $A/B$ 의 정수 부분을 나타냄

<68> 조건 3)  $\|C/B\| = \beta$ ,  $\|C/B\|$  는  $C/B$ 의 정수 부분을 나타냄

<69> 예 1) 일반적인 DVD의 경우

<70>  $\text{Data Frame 1} = 2K(b)$ ,  $\text{Data Frame 2} = 32K(B)$ , 최외주 두 개의 트랙내의 데이터 양 =  $284K(C)$ 일 때, 랜덤 데이터의 발생 주기는 최소  $17.75K(= 2K \times 284/32K)$  이상이어야 하고, 실제로 DVD의 스크램블러내의 랜덤 데이터 발생기의 주기는  $32K$  이다.

<71> 예 2) 트랙 방향의 선밀도가 DVD의 2배 정도인 HD-DVD의 제1 경우

<72>  $\text{Data Frame 1} = 4K(b)$ ,  $\text{Data Frame 2} = 64K(B)$ , 최외주 두 개의 트랙내의 데이터 양 =  $568K(C)$ 일 때, 랜덤 데이터의 발생 주기는 최소  $35.5K(= 4K \times 568/64K)$  이상이어야 한다.

<73> 예 3) 트랙 방향의 선밀도가 DVD의 2배 정도인 HD-DVD의 제2 경우

- <74> Data Frame 1 = 8K(b), Data Frame 2 = 64K(B), 최외주 두 개의 트랙내의 데이터 양 = 568K(C)일 때, 랜덤 데이터의 발생 주기는 최소  $71K(= 8K \times 568/64K)$  이상이어야 한다.
- <75> 예 4) 트랙 방향의 선밀도가 DVD의 2배 정도인 HD-DVD의 제3 경우
- <76> Data Frame 1 = 8K(b), Data Frame 2 = 128K(B), 최외주 두 개의 트랙내의 데이터 양 = 568K(C)일 때, 랜덤 데이터의 주기는 최소  $35.5K(= 8K \times 568/128K)$  이상이어야 한다.
- <77> 예 5) 트랙 방향의 선밀도가 DVD의 2배 정도인 HD-DVD의 제4 경우
- <78> Data Frame 1 = 2K(b), Data Frame 2 = 64K(B), 최외주 두 개의 트랙내의 데이터 양 = 568K(C)일 때, 랜덤 데이터의 주기는 최소  $17.75K(= 2K \times 568/64K)$  이상이어야 하고, 일반적인 DVD 시스템의 스크램블러를 사용할 수 있다.
- <79> 본 발명은 데이터 스크램블을 사용하는 기기에 사용가능하며, 특히 고밀도 디스크 시스템에 효율적으로 사용할 수 있다.

#### 【발명의 효과】

- <80> 본 발명의 스크램블러의 스크램블 방법은 고밀도 광기록재생기기에서 안정된 서보 신호 생성 및 변조시 DC 억압에 유리하다. 또한, 본 발명은 64K 이상의 긴 주기의 랜덤 데이터 발생이 가능하므로 HD-DVD 시스템에 적용할 수 있다.

**【특허청구범위】****【청구항 1】**

데이터 스크램블러에 있어서:

랜덤 데이터 발생 주기를 최소한 제1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 계산한 결과를 승산하여 얻어진 결과를 토대로하여 랜덤 데이터를 발생하는 랜덤 데이터 발생기를 가지는 스크램블러.

**【청구항 2】**

제1항에 있어서, 상기 제1 데이터 프레임 크기는 섹터이고, 제2 데이터 프레임 크기는 에러 정정 블록인 것을 특징으로 하는 스크램블러.

**【청구항 3】**

제1항에 있어서, 상기 랜덤 데이터 발생 주기는 최소한 제1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 계산한 결과를 승산하여 얻어진 값 이상인 것을 특징으로 하는 스크램블러.

**【청구항 4】**

제1항에 있어서, 상기 랜덤 데이터 발생기는,

직렬로 구성되며 n비트를 시프트 저장해서 랜덤 데이터를 발생하며, 최초의 초기 값을 포함하여 7번까지 시프트하여 출력되는 레지스터의 값들과 이들 값이 리턴하는 데 필요한 용량 직후의 값과 이 값으로부터 7번까지 시프트하여 출력되는 레지스터의 값들, 총 n개의 값을 초기화값으로 사용하는 레지스터; 및

직렬로 구성되며, 소정의 브랜치값의 유효한 브랜치의 수에 대응하는 상기 레지스

터의 출력과 인접한 논리 게이트의 출력을 배타적 논리합한 결과가 최하위 레지스터에 피드백 입력되는 복수개의 논리 게이트로 구성되는 제1 논리 회로를 포함하고,

상기 스크램블러는 상기 랜덤 데이터 발생기와, 소정수의 하위 레지스터의 출력과 입력 데이터를 바이트 단위로 배타적 논리합하여 스크램블 결과를 제공하는 제2 논리 회로를 포함하는 것을 특징으로 하는 스크램블러.

#### 【청구항 5】

제4항에 있어서, 상기  $n$ 을 16으로 하면 상기 랜덤 데이터의 주기는  $2^{16}(=64K)$ 가 되는 것을 특징으로 하는 스크램블러.

#### 【청구항 6】

제4항에 있어서, 상기 유효한 브랜치의 수는 4이상이고, 상기 유효한 브랜치의 값은 8016h, 801Ch, 8029h, 80D0h, 810Ah, 810Ch, 8112h, 8142h, 8148h, 8150h, 8214h, 8241h, 8244h, 8248h, 8260h, 8320h, 8406h, 8430h, 8540h, 8580h, 8610h, 8805h, 8821h, 8841h, 8842h, 8920h, 8940h, 8A04h, 9028h, 9082h, 9120h, 9420h, 9840h, 9C00h, A084h, A101h, A108h, A140h, A440h, A801h, A840h, B010h, B400h, C009h, C00Ah, C042h, C108h, C120h, C208h, C801h, CA00h, D008h 중 하나인 것을 특징으로 하는 스크램블러.

#### 【청구항 7】

제4항에 있어서, 상기 유효한 브랜치값이 'B400h'이고, 상기 초기값은 최초 0001h와 이를 7회까지 시프트 레프트한 값들(0002h, 0004h, 0008h, 0010h, 0020h, 0040h, 0080h)과 이들 값이 리턴하는 데 필요한 용량 32K이후의 레지스터의 결과인 7E80h와 이를 7회까지

시프트 레프트한 값들(FF01h, FE02h, FC04h, F808h, F011h, E023h, C046h)을 레지스터를 위한 초기값으로 사용하는 것을 특징으로 하는 스크램블러.

#### 【청구항 8】

제1항에 있어서, 상기 랜덤 데이터 발생기는

직렬로 구성되며 n비트를 시프트 저장해서 랜덤 데이터를 발생하며, 최초의 초기 값을 포함하여 4K번 시프트 레프트할 때마다 제공되는 레지스터의 값, 총 n개의 값을 초기값으로 사용하는 레지스터; 및

직렬로 구성되며, 소정의 브랜치값의 유효한 브랜치의 수에 대응하는 상기 레지스터의 출력과 인접한 논리 게이트의 출력을 배타적 논리합한 결과가 최하위 레지스터에 피드백 입력되는 복수개의 논리 게이트로 구성되는 제1 논리 회로를 포함하고,

상기 스크램블러는 상기 랜덤 데이터 발생기와, 소정수의 하위 레지스터의 출력과 입력 데이터를 비트 단위로 배타적 논리합하여 스크램블 결과를 제공하는 제2 논리 회로를 포함하는 것을 특징으로 하는 스크램블러.

#### 【청구항 9】

제8항에 있어서, 상기 n을 16으로 하면 랜덤 데이터의 주기는  $2^{16}(=64K)$ 가 되는 것을 특징으로 하는 스크램블러.

#### 【청구항 10】

제8항에 있어서, 상기 유효한 브랜치의 수는 4이상이고, 상기 유효한 브랜치의 값은 8016h, 801Ch, 8029h, 80D0h, 810Ah, 810Ch, 8112h, 8142h, 8148h, 8150h,

8214h, 8241h, 8244h, 8248h, 8260h, 8320h, 8406h, 8430h, 8540h, 8580h, 8610h, 8805h, 8821h, 8841h, 8842h, 8920h, 8940h, 8A04h, 9028h, 9082h, 9120h, 9420h, 9840h, 9C00h, A084h, A101h, A108h, A140h, A440h, A801h, A840h, B010h, B400h, C009h, C00Ah, C042h, C108h, C120h, C208h, C801h, CA00h, D008h 중 하나인 것을 특징으로 하는 스크램블러.

#### 【청구항 11】

제8항에 있어서, 상기 유효한 브랜치의 값은 'B400h'이고, 최초 0001h와 이로부터 매 4096회 시프트 레프트한 후 얻어지는 15개의 레지스터의 값들(3DADh, D4E7h, FDCAh, EBCCCh, 292Eh, 50F0h, BFCAh, 7F80h, D36Eh, BB39h, 5DFFh, A809h, 6647h, 8044h, 0304h)을 레지스터의 초기값으로 사용하는 것을 특징으로 하는 스크램블러.

#### 【청구항 12】

제1항에 있어서, 상기 랜덤 데이터 발생기는 제어값에 의해 제1 주기로 유효 브랜치가 달라지며, 제2 주기의 랜덤 데이터를 발생하는 것을 특징으로 하는 스크램블러.

#### 【청구항 13】

제12항에 있어서, 상기 제1 주기는 에러 정정 블록에 해당하고, 상기 제2 주기는 섹터에 해당하는 것을 특징으로 하는 스크램블러.

#### 【청구항 14】

제12항에 있어서, 상기 제2 주기는 4K에 해당하며, 제어값은 에러 정정 블록 단위로 829h, 834h, 84Ch, 868h, 883h, 891h, 8B0h, 8C2h, 906h, 960h, 990h, A03h, A18h, B04h, C48h, CA0h 중 하나를 제공하는 것을 특징으로 하는 것을 스크램블러.



**【청구항 15】**

제12항에 있어서, 상기 랜덤 데이터 발생기는

16 가지의 제어값에 대응하는 12비트의 출력 중 하나를 에러 정정 블록 단위로 제공하는 디코더;

직렬로 구성되며, 12비트를 시프트 저장하고 섹터 단위의 랜덤 데이터를 발생하는 레지스터;

상기 디코더로부터 제공되는 12비트의 출력을 선택 신호로 입력하여, 상기 디코더의 12비트의 출력 중 유효한 비트에 대해서는 '0'를 제공하고, 그렇지 않으면 상기 레지스터의 출력을 제공하는 선택 출력 회로; 및

상기 선택 출력 회로의 12비트의 출력과 상기 레지스터의 12비트의 출력을 배타적 논리합하되, 상기 디코더의 12비트의 출력 중 유효한 비트에 대해서만 배타적 논리합한 결과가 최하위 레지스터에 피드백 입력되는 제1 논리 회로를 포함하고,

상기 스크램블러는 랜덤 데이터 발생기와, 상기 랜덤 데이터 발생기와 소정수의 하위 레지스터의 출력과 입력 데이터를 바이트 단위로 배타적 논리합하여 스크램블 결과를 제공하는 제2 논리 회로를 포함하는 것을 특징으로 하는 스크램블러.

**【청구항 16】**

제15항에 있어서, 상기 레지스터의 초기값은 하나의 에러 정정 블록 단위로 새롭게 설정되는 것을 특징으로 하는 스크램블러.

**【청구항 17】**

랜덤 데이터 발생기를 이용하여 데이터를 스크램블하는 방법에 있어서:

상기 랜덤 데이터 발생기의 랜덤 데이터 발생주기를 최소한 제1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 계산한 결과를 승산하여 얻어진 결과를 토대로하여 랜덤 데이터를 발생하는 단계를 포함하는 스크램블 방법.

#### 【청구항 18】

제17항에 있어서, 상기 제1 데이터 프레임 크기는 섹터이고, 제2 데이터 프레임 크기는 여러 정정 블록인 것을 특징으로 하는 스크램블 방법.

#### 【청구항 19】

제17항에 있어서, 상기 랜덤 데이터 발생 주기는 최소한 제1 데이터 프레임 크기와, 최외주 두개의 트랙내의 데이터 양을 제2 데이터 프레임의 크기로 계산한 결과를 승산하여 얻어진 값 이상인 것을 특징으로 하는 스크램블 방법.

#### 【청구항 20】

제17항에 있어서,

상기 랜덤 데이터 발생 단계에서는 n비트를 시프트 저장해서 랜덤 데이터를 발생하며, 최초의 초기값을 포함하여 7번까지 시프트하여 출력되는 레지스터의 값들과 이들 값이 리턴하는 데 필요한 용량 직후의 값과 이 값으로부터 7번까지 시프트하여 출력되는 레지스터의 값들, 총 n개의 값을 초기값으로 사용하고,

소정수의 하위 레지스터의 출력과 입력 데이터를 바이트 단위로 배타적 논리합하여 스크램블 결과를 제공하는 단계를 더 포함하는 스크램블 방법.

**【청구항 21】**

제17항에 있어서,

상기 랜덤 데이터 발생 단계에서는 n비트를 시프트 저장해서 랜덤 데이터를 발생하며, 최초의 초기값을 포함하여 4K번 시프트 레프트할 때마다 제공되는 레지스터의 값, 총 n개의 값을 초기값으로 사용하고,

소정수의 하위 레지스터의 출력과 입력 데이터를 비트 단위로 배타적 논리합하여 스크램블 결과를 제공하는 단계를 더 포함하는 스크램블 방법.

**【청구항 22】**

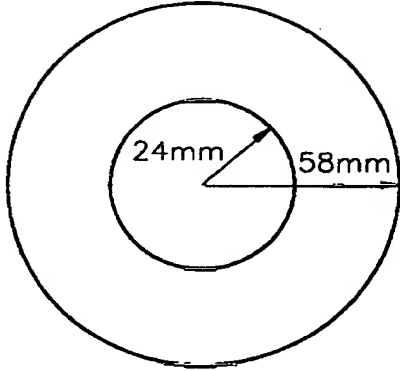
제17항에 있어서,

상기 랜덤 데이터 발생기 단계에서는 에러 정정 블록 단위로 제공되는 16가지의 제어값에 대응하여 섹터 단위의 랜덤 데이터를 발생하는 레지스터의 초기값은 하나의 에러 정정 블록 단위로 새롭게 설정하고,

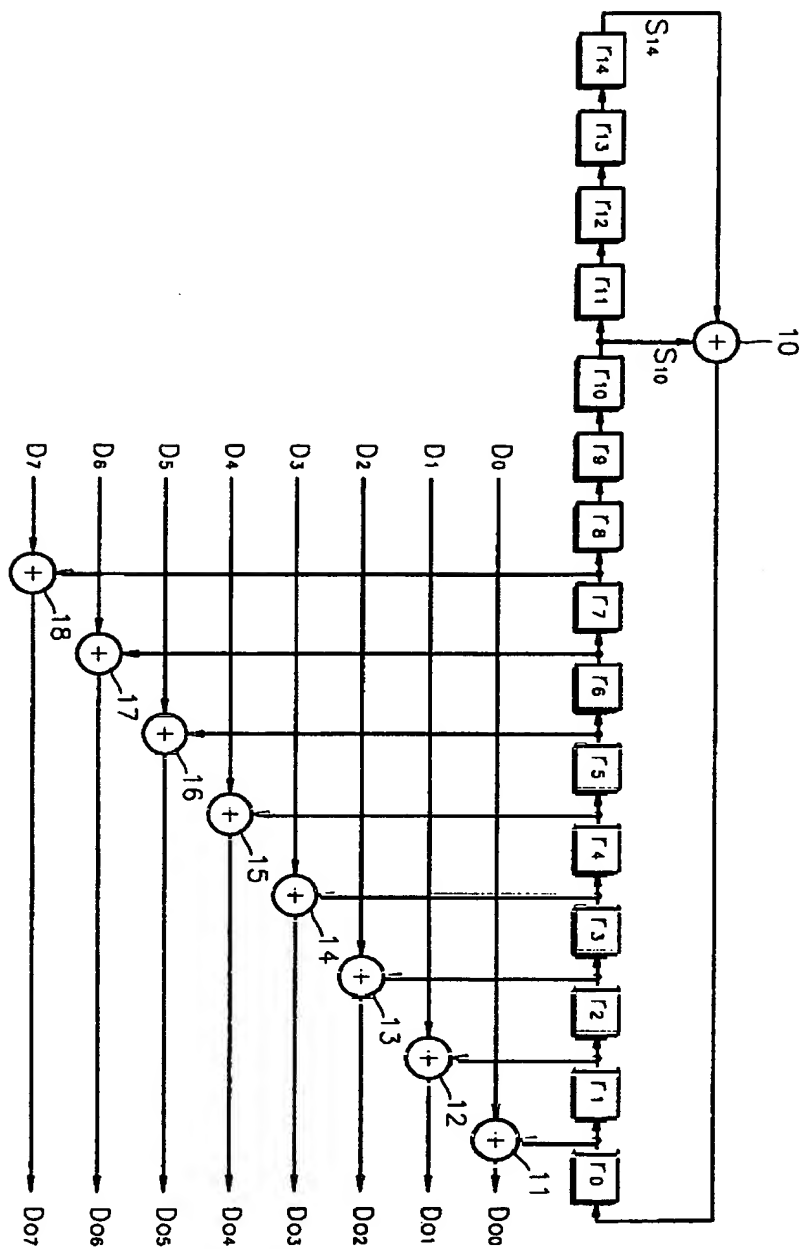
소정수의 하위 레지스터의 출력과 입력 데이터를 바이트 단위로 배타적 논리합하여 스크램블 결과를 제공하는 단계를 더 포함하는 스크램블 방법.

【도면】

【도 1】



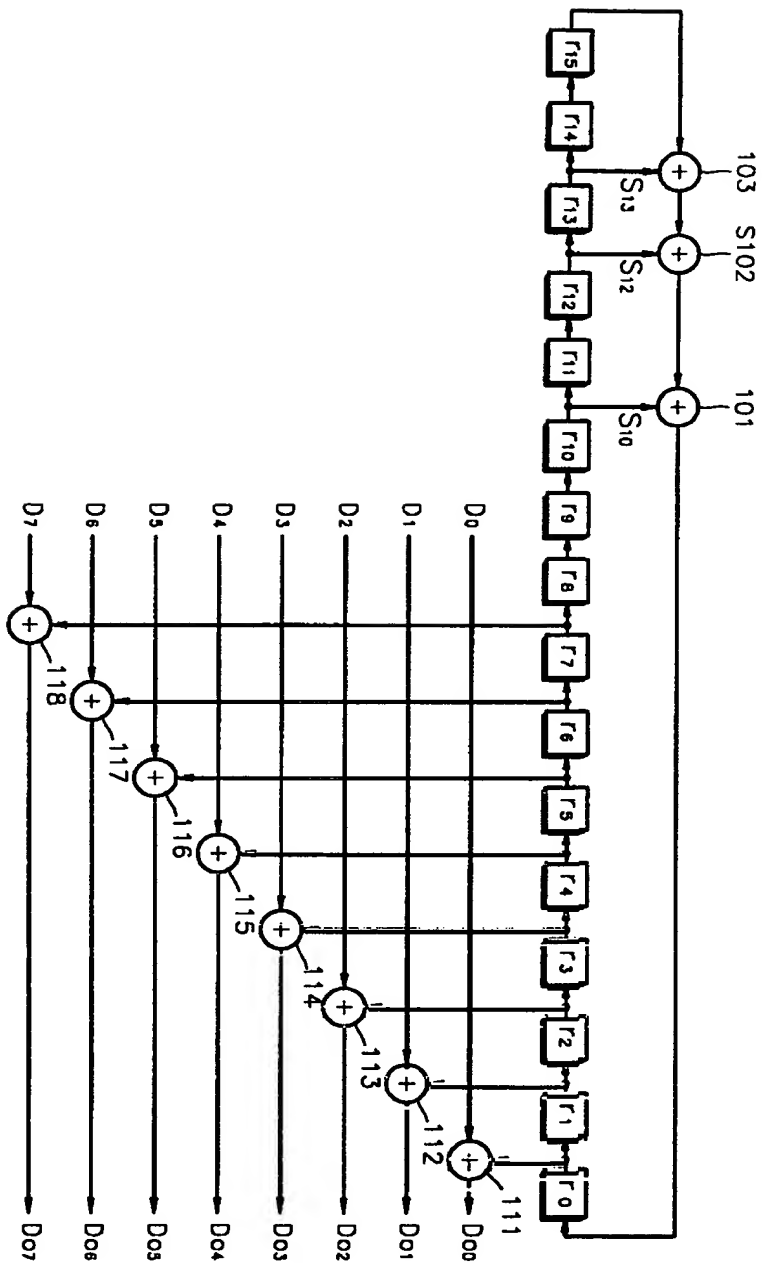
【图 2】



【도 3】

| ID(7:4) | 초기값   | ID(7:4) | 초기값   |
|---------|-------|---------|-------|
| 0h      | 0001h | 8h      | 0010h |
| 1h      | 5500h | 9h      | 5000h |
| 2h      | 0002h | Ah      | 0020h |
| 3h      | 2A00h | Bh      | 2001h |
| 4h      | 0004h | Ch      | 0040h |
| 5h      | 5400h | Dh      | 4002h |
| 6h      | 0008h | Eh      | 0080h |
| 7h      | 2800h | Fh      | 0005h |

【도 4】

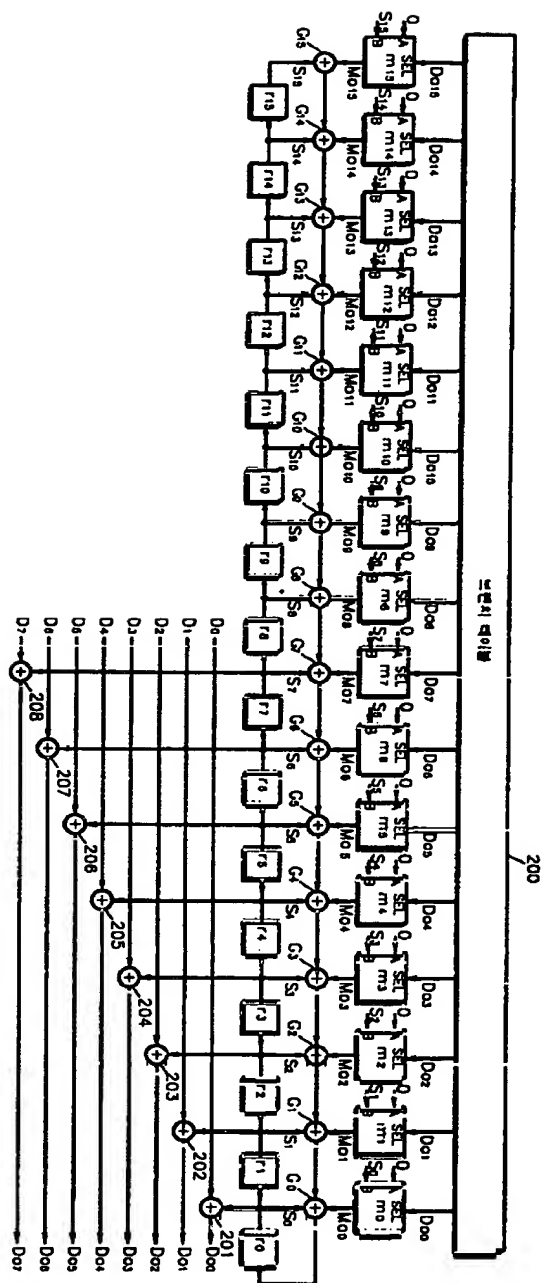


【도 5】

| ID(7:4) | 초기값   | ID(7:4) | 초기값   |
|---------|-------|---------|-------|
| 00h     | 0001h | 08h     | 0010h |
| 01h     | 7F80h | 09h     | F808h |
| 02h     | 0002h | 0Ah     | 0020h |
| 03h     | FF01h | 0Bh     | F011h |
| 04h     | 0004h | 0Ch     | 0040h |
| 05h     | FE02h | 0Dh     | E023h |
| 06h     | 0008h | 0Eh     | 0080h |
| 07h     | FC04h | 0Fh     | C046h |



【도 6】



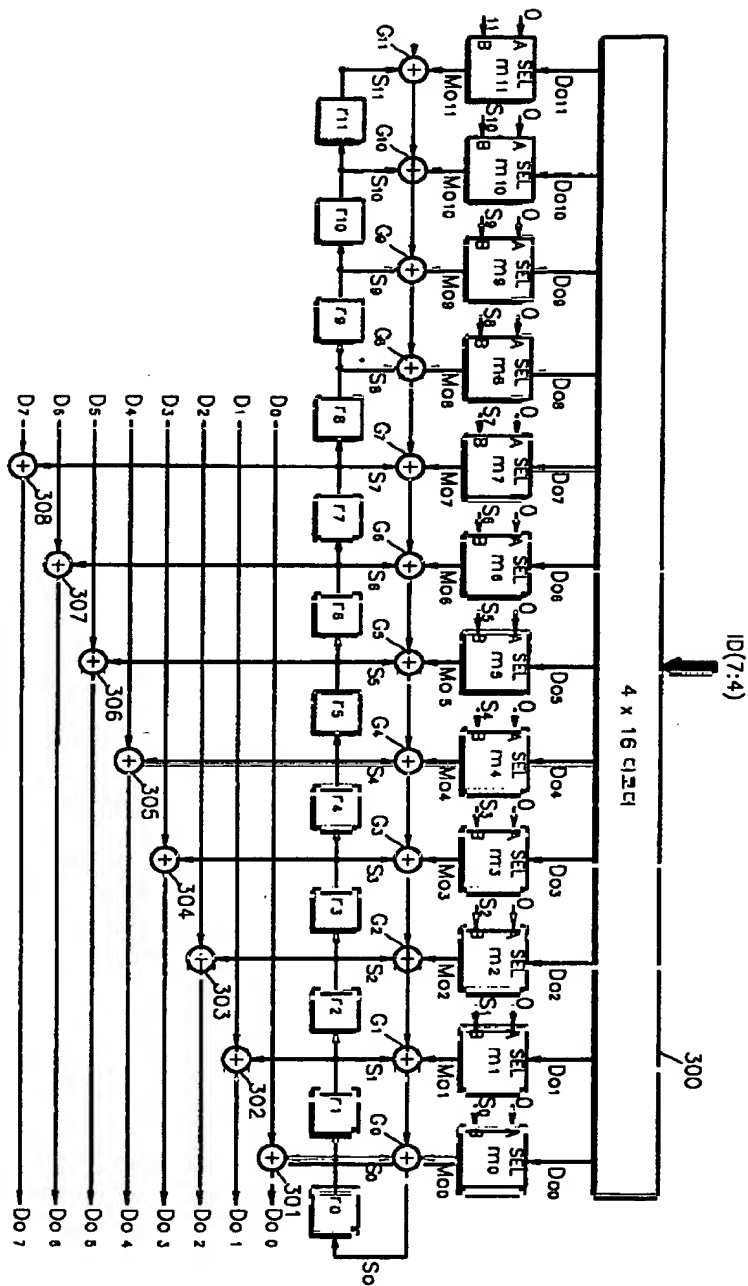
【도 7】

| NO | branch지<br>(Do15 ~ Do0) | NO | branch지<br>(Do15 ~ Do0) |
|----|-------------------------|----|-------------------------|
| 1  | 8016h                   | 27 | 8940h                   |
| 2  | 801Ch                   | 28 | 8A04h                   |
| 3  | 8029h                   | 29 | 9028h                   |
| 4  | 80D0h                   | 30 | 9082h                   |
| 5  | 810Ah                   | 31 | 9120h                   |
| 6  | 810Ch                   | 32 | 9420h                   |
| 7  | 8112h                   | 33 | 9840h                   |
| 8  | 8142h                   | 34 | 9C00h                   |
| 9  | 8148h                   | 35 | A084h                   |
| 10 | 8150h                   | 36 | A101h                   |
| 11 | 8214H                   | 37 | A108h                   |
| 12 | 8241H                   | 38 | A140h                   |
| 13 | 8244H                   | 39 | A440h                   |
| 14 | 8248H                   | 40 | A801h                   |
| 15 | 8260H                   | 41 | A840h                   |
| 16 | 8320H                   | 42 | B010h                   |
| 17 | 8406H                   | 43 | B400h                   |
| 18 | 8430H                   | 44 | C009h                   |
| 19 | 8540H                   | 45 | C00Ah                   |
| 20 | 8580H                   | 46 | C042h                   |
| 21 | 8610H                   | 47 | C108h                   |
| 22 | 8805H                   | 48 | C120h                   |
| 23 | 8821H                   | 49 | C208h                   |
| 24 | 8841H                   | 50 | C801h                   |
| 25 | 8842H                   | 51 | CA00h                   |
| 26 | 8920H                   | 52 | DO08h                   |

【도 8】

| ID(7:4) | 초기값   | ID(7:4) | 초기값   |
|---------|-------|---------|-------|
| 00h     | 0001h | 08h     | 7F80h |
| 01h     | 3DADh | 09h     | D36Eh |
| 02h     | D4E7h | 0Ah     | BB39h |
| 03h     | FDCAh | 0Bh     | 5DFFh |
| 04h     | EBCCh | 0Ch     | A809H |
| 05h     | 292Eh | 0Dh     | 6647h |
| 06h     | 50F0h | 0Eh     | 8044h |
| 07h     | BFCAh | 0Fh     | 0304h |

【도 9】



【도 10】

| ID(7:4) | 초기값  | ID(7:4) | 초기값  |
|---------|------|---------|------|
| 00h     | 829h | 08h     | 906h |
| 01h     | 834h | 09h     | 960h |
| 02h     | 84Ch | 0Ah     | 990h |
| 03h     | 868h | 0Bh     | A03h |
| 04h     | 883h | 0Ch     | A18H |
| 05h     | 891h | 0Dh     | B04h |
| 06h     | 8B0h | 0Eh     | C48h |
| 07h     | 8C2h | 0Fh     | CA0h |